AUTONOMIC INFRASTRUCTURE ENABLEMENT FOR POINT IN TIME COPY CONSISTENCY GROUPS

CROSS-REFERENCED APPLICATIONS

[1] This application incorporates by reference commonly-assigned and co-pending U.S. Patent Serial No. 10/464,024, filed June 6, 2003, and entitled METHOD, SYSTEM AND ARTICLE OF MANUFACTURE FOR REMOTE COPYING OF DATA. This application also incorporates by reference commonly-assigned and co-pending Docket Nos. TUC9-2003-0099US1, entitled METHOD, SYSTEM, AND PROGRAM FOR RECOVERY FROM A FAILURE IN AN ASYNCHRONOUS DATA COPYING SYSTEM; TUC9-2003-0045US1, entitled APPARATUS AND METHOD TO COORDINATE MULTIPLE DATA STORAGE AND RETREIVAL STORAGE SYSTEMS; TUC9-2003-0108US1, entitled METHOD, SYSTEM AND PROGRAM FOR FORMING A CONSISTENCY GROUP; TUC9-2003-0116US1 entitled METHOD, SYSTEM AND ARTICLE OF MANUFACTURE FOR RECOVERY FROM A FAILURE IN A CASCADING PPRC SYSTEM; TUC--2003-0100US1, entitled METHOD, SYSTEM, AND PROGRAM FOR MIRRORING DATA AMONG STORAGE SITES; and TUC9-2003-0119US1, entitled METHOD, SYSTEM AND PROGRAM FOR ASYNCHRONOUS COPY, all filed on September 29, 2003.

TECHNICAL FIELD

[2] The present invention relates generally to data backup in a data storage system and, in particular, to a method, system and computer program product for protecting consistency groups during a virtual copy operation, such as an IBM developed FlashCopy®, in an asynchronous peer-to-peer remote copy system.

BACKGROUND ART

[3] Information technology systems, including storage systems, may need protection from site disasters or outages, where outages may be planned or unplanned. Furthermore, information technology systems may require features for data migration, data backup, or data duplication. Implementations for disaster or outage

recovery, data migration, data backup, and data duplication may include mirroring or copying of data in storage systems. Such mirroring or copying of data may involve interactions among hosts, storage systems and connecting networking components of the information technology system.

[4] A storage server, such as the IBM® TotalStorage® Enterprise Storage Server® ("ESS"), may be a disk storage server that includes one or more processors coupled to storage devices, including high capacity scalable storage devices, Redundant Array of Inexpensive (or Independent) Disks ("RAID"), etc. The enterprise storage servers are connected to a network and include features for copying data in storage systems.

[5] Peer-to-Peer Remote Copy ("PPRC") is an ESS function that allows the shadowing of application system data from a first site to a second site. The first site may be referred to as an application site, a local site, or a primary site. The second site may be referred to as a recovery site, a remote site or a secondary site. The logical volumes that hold the data in the ESS at the primary site are called primary volumes, and the corresponding volumes that hold the mirrored data at the secondary site are called secondary volumes. High speed links, such as IBM ESCON® links may connect the primary and secondary ESS systems.

[6] In the synchronous type of operation for PPRC, i.e., synchronous PPRC, the updates done by a host application to the primary volumes at the primary site are synchronously shadowed onto the secondary volumes at the secondary site. As synchronous PPRC is a synchronous copying solution, write updates are ensured on both copies (primary and secondary) before the write is considered to be completed for the host application. In synchronous PPRC the host application does not get the "write complete" response until the update is synchronously done in both the primary and the secondary volumes. Therefore, from the perspective of the host application, the data at the secondary volumes at the secondary site is equivalent to the data at the primary volumes at the primary site.

[7] Inherent to synchronous PPRC operations is an increase in the response time as compared to asynchronous copy operation. The overhead comes from the additional steps which are executed before the write operation is signaled as

completed to the host application. Also, the PPRC activity between the primary site and the secondary site may be comprised of signals and data which travel through the links that connect the sites, and the overhead response time of the host application write operations will increase proportionally with the distance between the sites. Therefore, the distance affects a host application's response time. In certain implementations, there may be a maximum supported distance for synchronous PPRC operations referred to as the synchronous communication distance.

[8] In the Extended Distance PPRC method of operation, PPRC mirrors the updates of the primary volume onto the secondary volumes in an asynchronous manner, while the host application is running. In asynchronous PPRC, the host application receives a write complete response before the update is copied from the primary volumes to the secondary volumes and a host application's write operations are free of the typical synchronous overheads. Therefore, asynchronous PPRC is suitable for secondary copy solutions at very long distances with minimal impact on host applications. However, asynchronous PPRC does not continuously maintain a consistent (point-in-time) copy of the primary data at the secondary site, therefore risking data loss in certain circumstances.

SUMMARY OF THE INVENTION

[9] The present invention provides methods, apparatus and computer program product for protecting consistency groups during data storage backup operations, particularly during the FlashCopy operation to create a new consistency group. The method includes two phases. In the first phase, a write-inhibit flag or indicator is imposed on FlashCopy source volumes of a new consistency group as part of their preparation for the FlashCopy operation. If the preparation of any volumes of the new consistency group are unsuccessful, the FlashCopy is withdrawn and the prior consistency group is retained, thereby aborting the formation of the new point-in-time copy (consistency group) and preventing corruption of the prior consistency group. If all volumes in the consistency group are successfully prepared, then in the second phase the write-inhibit flags are released and the FlashCopy operation

committed, thereby indicating that the new consistency group has been secured. PPRC write requests may then resume to the secondary PPRC volumes.

[10] The apparatus includes FlashCopy source and target devices. Means are included to impose, in a first FlashCopy phase, FlashCopy source volumes of a new consistency group are prepared for the FlashCopy operation, including imposing a write-inhibit flag on the FlashCopy source volumes. Means are further included to determine if the preparation of any volumes are unsuccessful. If so, means are included to execute a withdraw the FlashCopy and the prior FlashCopy point-in-time copy is retained, thereby preventing the corruption of the prior consistency group. If all FlashCopy source volumes in the current consistency group are successfully prepared, means are included to, in a second phase, execute a commit command whereby the FlashCopy operation is committed and the FlashCopy source volumes of the new consistency group have their write-inhibit flags removed, thereby indicating that the new consistency group has been secured. PPRC write requests may then resume to the secondary PPRC volumes.

[11] The computer program product includes computer-readable code comprising a two-phase set of instructions for performing a FlashCopy operation. In the first phase, a write-inhibit flag or indicator is imposed on FlashCopy source volumes of a new consistency group as part of their preparation for the FlashCopy operation. If the preparation of any volumes of the new consistency group are unsuccessful, the FlashCopy is withdrawn and the prior consistency group is retained, thereby aborting the formation of the new point-in-time copy (consistency group) and preventing corruption of the loss prior consistency group. If all volumes in the consistency group are successfully prepared, then in the second phase the write-inhibit flags are released and the FlashCopy operation committed, thereby indicating that the new consistency group has been secured. PPRC write requests may then resume to the secondary PPRC volumes.

BRIEF DESCRIPTION OF THE DRAWINGS

[12] Referring now to the drawings in which like reference numbers represent corresponding elements throughout:

[13] Fig. 1 is a block diagram of a network computing environment in which aspects of the invention may be implemented;

[14] Fig. 2 is a block diagram of asynchronous data transfer and FlashCopy applications in accordance with certain described implementations of the invention; and

[15] Fig. 3 is a flowchart illustrating logic in accordance with certain described implementations of the invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[16] In the following description, reference is made to the accompanying drawings which form a part hereof and which illustrate several implementations. It is understood that other implementations may be utilized and structural and operational changes may be made without departing from the scope of the present limitations.

[17] FIG. 1 illustrates a computing environment utilizing two storage control units, such as a primary storage control unit 102~~,~~ and a secondary storage control unit 104 connected by a data interface channel~~s~~ 108, such as the ESCON channel or any other data interface mechanism known in the art (e.g., fibre channel, Storage Area Network (SAN) interconnections, etc.). The two storage control units 102 and 104 may be at two different sites and asynchronously interconnected. Additionally, the secondary storage control unit 104 may be in a secure environment separated from the primary storage control unit 102 and with separate power to reduce the possibility of an outage affecting both the primary storage control unit 102 and the secondary storage control unit 104.

[18] Fig. 2 illustrates a network computing environment 200 in which aspects of the present invention may be implemented. In such an environment, the primary storage control unit 102, along with the primary storage volumes 116, may be among several (or many) storage controllers and storage volumes at a local site or sites 210. Similarly, the secondary storage control unit 104, along with the secondary storage volumes 118 and 120, may be among several (or many) storage controllers and storage volumes at a remote site or sites 212.

[19] Referring back to Fig. 1, the primary storage control unit 102 is coupled to a host 111 via data interface channel 112. While only a single host 111 is shown coupled to the primary storage control unit 102, a plurality of hosts may be coupled to the primary storage control unit 102. The host 111 may be any computational device known in the art, such as a personal computer, a workstation, a server, a mainframe, a hand held computer, a telephony device, a network appliance, etc. The host 111 may include any operating system (not shown) known in the art, such as the IBM OS/390® operating system. The host 111 may include at least one host application 114 that sends Input/Output (I/O) requests (including write requests) to the primary storage control unit 102.

[20] The storage control units 102 and 104 are coupled to storage volumes such as primary site storage volumes 116 and secondary site storage volumes 118 and 120, respectively. The storage volumes 116, 118, 120 may be configured as a Direct Access Storage Device (DASD), one or more RAID ranks, just a bunch of disks (JBOD), or any other data repository system known in the art. The storage control units 102 and 104 may each include a cache, such as caches 122 and 124 respectively. The caches 122 and 124 comprise volatile memory to store data blocks (for example, formatted as tracks). The storage control units 102 and 104 may each include a non-volatile storage (NVS), such as non-volatile storage 128 and 130 respectively. The non-volatile storage 128 and 130 elements may buffer certain modified data blocks in the caches 122 and 124 respectively.

[21] The primary storage control unit 102 additionally includes an application, such as a primary PPRC application 134, for asynchronous copying of data stored in the cache 122, non-volatile storage 128 and primary site storage volumes 116 to another storage control unit, such as the secondary storage control unit 104. The primary PPRC application 134 includes functions which execute in the primary storage control unit 102. The primary storage control unit 102 receives I/O requests from the host application 114 to read and write from and to the primary site storage volumes 116.

[22] The secondary storage control unit 104 additionally includes an application such as a secondary PPRC application 136. The secondary PPRC application 136

includes functions that execute in the secondary storage control unit 104. The secondary PPRC application 136 can interact with the primary storage control unit 102 to receive data asynchronously. A FlashCopy application 140 includes functions which ensure that until a data block in a FlashCopy relationship has been hardened to its location on a target disk, the data block resides on the source disk. The FlashCopy application 140 includes functions which execute Flashcopy-revertible, Withdraw-FlashCopy-revert and Withdraw-FlashCopy-commit commands of this invention.

[23] Therefore, FIG. 1 illustrates a computing environment in which a host application 114 sends I/O requests to a primary storage control unit 102. The primary storage control unit 102 asynchronously copies data to the secondary storage control unit 104, and the secondary storage control unit 104 subsequently copies data from the PPRC Secondary storage volumes (FlashCopy Source volumes) 118 to the FlashCopy Target volumes 120 during FlashCopy operations. As a result of asynchronous copying, the effect of long distance on the host response time is eliminated.

[24] The logic for processing a write request from the host application 114 will be described briefly. Control begins when the primary PPRC application 134 receives a write request from the host application 114. The primary PPRC application 134 writes data corresponding to the write request on the cache 122 and the non-volatile storage 128 on the primary storage control unit 102. Once the data is stored in the cache 122 and NVS 128, the primary PPRC application 134 signals to the host application 114 that the write request from the host application 114 has been completed at the primary storage control unit 102. The primary PPRC application 134 may then receive a next write request from the host application 114. Additional applications (not shown), such as caching applications and non-volatile storage applications, in the primary storage control unit 102 may manage the data in the cache 122 and the data in the non-volatile storage 128 and keep the data in the cache 122 and the non-volatile storage 128 consistent with the data in the primary site storage volumes 116.

[25] Because the secondary storage control unit 104 receives data updates asynchronously, the volumes 118 on the secondary storage control unit 106 may not be consistent with the volumes 116 on the primary storage control unit 102. However, all of the volumes 118 from the secondary storage control unit 104 will be consistent at certain points in time. The consistent set of volumes (FlashCopy source volumes, also referred to as a "consistency group") at the secondary storage control unit 106 may then be preserved via a point-in-time FlashCopy to the FlashCopy target volumes 120. Preferably, writes to the secondary storage control unit 104 are inhibited or otherwise prevented between the primary and secondary control units 102 and 104, while the secondary storage control unit 104 catches up with the updates. Once the FlashCopy is completed to the FlashCopy volumes 120, recovery from a subsequent failure in the primary or secondary units 102 or 104 may be possible by restoring to the prior point-in-time consistency group stored on the FlashCopy target volumes 120.

[26] As illustrated in Fig. 2, the consistency group may be distributed over many storage volumes in many storage controllers. Due to the distributed nature of the environment, FlashCopy commands to the source volumes (on the secondary unit 104) do not execute simultaneously. Consequently, once a FlashCopy operation begins, the FlashCopy target volumes 120 become inconsistent until the FlashCopy of all source volumes 118 is completed: some source/target volume pairs may have completed the FlashCopy, others may be in process of executing the FlashCopy command and still other may not have received the command yet. This period of time is the FlashCopy transition phase and, as long as no write request is received by the FlashCopy source (PPRC secondary) volumes 118, a reversion to the prior consistent set of volumes is still possible; that is, the prior consistency group will remain intact.

[27] However, during the transition phase, the asynchronous PPRC mechanism of the primary unit 102 may time out, perform a warmstart or otherwise cause the I/O to the secondary unit 104 to resume. In such an event, the FlashCopy source volumes are no longer consistent and, because the FlashCopy target volumes are also inconsistent, reversion to the prior FlashCopy consistency group is not possible.

This presents a window during which a disaster or failure at the primary unit 102 exposes data consistency to loss, particularly if the FlashCopy operation for any, but not all, of the volumes is unsuccessful. The present invention provides a method and means for reducing or eliminating such risk as will now be described with reference to Fig. 3.

[28] Table I represents a configuration of volumes and their relationships of volumes in the PPRC primary 116, the PPRC secondary (FlashCopy source) 118 and the FlashCopy target 120. Table I will be described in conjunction with Fig. 3, a flowchart of an implementation of the present invention.

TABLE I

PPRC Primary	PPRC Secondary/ FlashCopy Source	FlashCopy Target
Volume A1	Volume B1	Volume C1
Volume A2	Volume B2	Volume C2
Volume A3	Volume B3	Volume C3
Volume A4	Volume B4	Volume C4
Volume A5	Volume B5	Volume C5

[29] Volumes C1-C5 in the FlashCopy target 120 contain an older ("prior") consistency group (represented by step 300). Next, data updates are transferred from the host application 114 to primary PPRC volumes A1-A5 116, respectively, on the primary storage control unit 102 and represent a "new" consistency group (step 302). The new consistency group is transferred from the PPRC primary volumes 116 to the FlashCopy source volumes B1-B5 118, respectively, in the secondary storage control unit 104 (step 304). (It will be appreciated that any number of volumes may be transferred; five has been arbitrarily selected herein for descriptive purposes only and not by way of limitation.) The FlashCopy target volumes C1-C5 120 continue to contain the prior consistency group. It is now desired to FlashCopy from the B volumes to the C volumes. An Establish-FlashCopy-revertable command

is generated to prepare the volume B1 in the secondary storage control unit 104 for a point-in-time copy operation. As the volume is prepared, the FlashCopy source volume B1 is write-inhibited (step 310). An attempt is made to prepare and write-inhibit the next source volume (B2) (step 312). The process continues (steps 314-318) until an attempt has been made to prepare and write-inhibit all source volumes B1-B5 to FlashCopy to the FlashCopy target volumes C1-C5.

[30] Due to the write-inhibit indicators associated with successfully prepared source volumes, if any failures occur in any primary control units containing volumes corresponding to the source volumes and a primary control unit attempts to transmit a write request to the secondary storage control unit (primary PPRC) 104, the write-inhibit flag will cause the write request to fail and neither the prior consistency group C1-C5 nor the new consistency group B1-B5 will be corrupted.

[31] If any FlashCopy preparations have been unsuccessful (step 320), such as volume B3 (due perhaps to a communication failure), then the successfully prepared volumes (those with write-inhibit indicators, volume pairs B1/C1, B2/C2, B4/C4 and B5/C5) are reverted with a Withdraw-FlashCopy-revert command (step 322); any volumes which failed in the preparation operation (volume B3/C3) remain unchanged (retaining the prior contents). As a result, the FlashCopy target volumes C1-C5 retain the prior consistency group in uncorrupted form. On the other hand, if the preparation operations of all source volumes were successful, a Withdraw-FlashCopy-commit command is generated to remove the write-inhibit indicators (step 324), signifying that formation of the current consistency group has been complete and secured in FlashCopy target volumes C1-C5. New writes may then be processed by the PPRC primary and secondary units. Moreover, any failures in the primary storage control unit 102 which cause data to attempt to flow from the PPRC primary A volumes 116 to the PPRC secondary B volumes 118 will fail and not corrupt the FlashCopy operation.

[32] In a variation of the foregoing procedure, a determination may be made following each attempt to prepare a FlashCopy source volume as to whether the preparation was successful. If the preparation was unsuccessful, the procedure may jump to step 322 and the Withdraw-FlashCopy-revert command issued to abort the

FlashCopy operation. Alternatively, the determination may be made following completion of the attempts to prepare all FlashCopy source volumes.

[33] Thus, a FlashCopy operation is a two-phase process. In the first phase ("prepare"), each FlashCopy is made "revertable" by write-inhibiting the source volume (with the Establish-FlashCopy-revertable command). If any FlashCopy preparation fails, the withdraw-FlashCopy-revert command may be executed, thereby causing the prior consistency group to remain intact. In the second phase, executed if all FlashCopy preparations are successful, the Withdraw-FlashCopy-commit command may be executed to remove all write-inhibit indicators and allow write requests from the primary control unit to resume.

[34] The described techniques may be implemented as a method, apparatus or article of manufacture using standard programming and/or engineering techniques to produce software, firmware, hardware, or any combination thereof. The term "article of manufacture" as used herein refers to code or logic implemented in hardware logic (e.g., an integrated circuit chip, Programmable Gate Array (PGA), Application Specific Integrated Circuit (ASIC), etc.) or a computer readable medium (e.g., magnetic storage medium such as hard disk drives, floppy disks, tape), optical storage (e.g., CD-ROMs, optical disks, etc.), volatile and non-volatile memory devices (e.g., EEPROMs, ROMs, PROMs, RAMs, DRAMs, SRAMs, firmware, programmable logic, etc.). Code in the computer readable medium is accessed and executed by a processor. The code in which implementations are made may further be accessible through a transmission media or from a file server over a network. In such cases, the article of manufacture in which the code is implemented may comprise a transmission media such as network transmission line, wireless transmission media, signals propagating through space, radio waves, infrared signals, etc. Of course, those skilled in the art will recognize that many modifications may be made to this configuration without departing from the scope of the implementations and that the article of manufacture may comprise any information bearing medium known in the art.

[35] In certain implementations, data in the storage devices is arranged in volumes. In alternative implementations, other storage unit values may be assigned; such

storage units may comprise tracks in a volume, blocks, logical subsystems, logical drives, or any other physical or logical storage unit designation known in the art.

[36] The illustrated logic of the Figs. show certain events occurring in a certain order. In alternative implementations, certain operations may be performed in a different order, modified or removed. Moreover, steps may be added to the above described logic and still conform to the described implementations. Further, operations described herein may occur sequentially or certain operations may be processed in parallel. Yet further, operations may be performed by a single processing unit or by distributed processing units.

[37] The objects of the invention have been fully realized through the embodiments disclosed herein. Those skilled in the art will appreciate that the various aspects of the invention may be achieved through different embodiments without departing from the essential function of the invention. The particular embodiments are illustrative and not meant to limit the scope of the invention as set forth in the following claims.